System and Method for a Made to Specification e-commerce Quoting and Orders Processing System on a Stand Alone or Integrated Portal

Claims

What is claimed:

1. E-commerce system specifications for defining product specifications with or without option variable questions via a XML Schema Definition-XSD file indicated is a given implementation of just one schema and any other applicable schema's file name can be used in other implementation such as Document Type Definition-DTD files, or other new industry schemas as defined in claims 2-25.

2. The method of claim 1, wherein each title of data input for each of the XSD elements see **FIG. 3** line 11 <title>1. Width Inches</title>.

3. The method of claim 1, wherein more information links see **FIG. 3** line 16 <helpURL>./MoreInfo/M2specsXML339950-2201-001.html#width</helpURL>.

4. The method of claim 1, wherein the width of the screen for claim 3 See **FIG. 3** Line 17 <helpwidth>700</helpwidth>.

5. The method of claim 1, wherein the height of the screen for claim 3 See **FIG. 3** Line 18 <helpheight>350</helpheight>.

6. The method of claim 1, wherein the title of the screen for for claim 3 See FIG. 3 Line 19 <helptitle>More Info For 1. Width Window Inches </helptitle>.

7. The method of claim 1, wherein the format ability of Displaying the prompt Question, input, & more information link on the following conditions.

8. The method of claim 7, wherein <prompt>NewLine</prompt> with all on the same line as default See **FIG. 3** line 20.

9. The method of claim 7, wherein multiple elements on the same line = <prompt>SameLine</prompt>.

10. The method of claim 7, wherein the prompt is on one line and input/more information link on the next line = <prompt>NewLine</prompt>.

11. The method of claim 7, wherein the ability for a system parameter in the web.config ASCII file to specify percentages of the line each component will occupy for claim 2 and 3.

12. The method of claim 11, wherein the "40:60/40:50:10" notation means 'if there is no more info, make a 40% prompt and 60% option; if there *is* a more info link, make a 40% prompt, then 50% option, then 10% more info'.

13. The method of claim 11, wherein similarly for the same line case, where there are four/five numbers instead of two/three, because there are two prompt/options on a line, and for the NewLine case, when there is no width for the prompt string, just for the option (and possibly the more info link).

14. The method of claim 11, wherein <add key="PSC_ColSpan_Normal" value="50:50/40:50:10" />.

15. The method of claim 11, wherein <add key="PSC_ColSpan_SameLine"value="25:25:25:25/20:25:20:25:10" />.

16. The method of claim 11, wherein <add key="PSC_ColSpan_NewLine" value="100/90:10" />.

17. The method of claim 1, wherein type of data input errors occur see **FIG. 3** Line 37 <xs:restriction base="xs:integer"> Other types are string, decimal, & any allowed type in XSD.

18. The method of claim 17, wherein Edit rules allowed on the answer.

19. The method of claim 18, wherein Edit rule Low = see **FIG. 4** Line 38 <xs:minInclusive value="6" />.

20. The method of claim 18, wherein Edit rule Max = see **FIG. 4** Line 43 <xs:maxInclusive value="6" />.

21. The method of claim 18, wherein Edit rule High = <xs:maxLength value="25" />.

22. The method of claim 18, wherein Edit rule Patern = <xs:pattern value="[0-9]+(\.[0-9]+)?" /> Any allowed type in XSD.

23. The method of claim 18, wherein Edit rule of any allowed type in XSD.

24. The method of claim 1, wherein error message words if input errors occur see **FIG. 3** Line 15 <errormessage>1. Width Must be Minimum 6, no Max. </errormessage>.

25. The method of claim 1, wherein Parameter to indicate if the price of an options drop down is displayed see **FIG. 4** Line 49 <display-price>True</display-price>.

26. The method of claim 1-25 wherein that by repeating the elements to achieve unlimited elements option specifications product option questions.

27. The method of claim 1 wherein that option selections will have user definable checking within all elements.

28. The method of claim 27, wherein the option selections will have user definable checking within all elements. See **FIG. 3** Line 12-14

```
<relations>
     <exact element="CouponID" value="psc#1" errormessage=" 12. Your coupon ID
is not valid"/>
</relations>
```

Type: Six 6 allow types.

29. The method of claim 27, wherein **minInclusive - a low boundary limit which may be reached** the minInclusive element is used to specify a minimum valid value for another element of the XML document. This applies to both numeric values and enumerations; in the latter case, the acceptable range is the specified value and all following it in the definition of the enumeration.

30. The method of claim 27, wherein **minExclusive - a low boundary limit which may not be reached** the minExclusive element

is used to specify a minimum value for another element of the XML document, but the value specified is NOT acceptable. This applies to both numeric values and enumerations; in the latter case, the acceptable range consists of all values following the specified one in the definition of the enumeration.

31. The method of claim 27, wherein **maxInclusive - a high boundary limit which may be reached** the maxInclusive element

is used to specify a maximum valid value for another element of the XML document. This applies to both numeric values and enumerations; in the latter case, the acceptable range is the specified value and all preceding it in the definition of the enumeration.

32. The method of claim 27, wherein **maxExclusive - a high boundary limit which may not be reached** the maxExclusive element

is used to specify a maximum value for another element of the XML document, but the value specified is NOT acceptable. This applies to both numeric values and enumerations; in the latter case, the acceptable range consists of all values preceding the specified one in the definition of the enumeration.

33. The method of claim 27, wherein **exact - an exact value** the exact **element is used to** specify an exact value for another element of the XML

document. This applies to all non-composite XSD types: numeric types, strings, and enumerations.

34. The method of claim 27, wherein **pattern - a regular expression pattern match** The pattern element is used to specify a regular expression pattern that another element of the XML document should match. This applies to all XSD types derived from the string type.

35. E-commerce system specifications for defining product specifications Via XSL file the ability to define; math to compute fields back into the e-commerce product.

36. The method of claim 35, wherein Selling price see **FIG. 8** Line 23, 30 thru line 32 <price><xsl:value-of select="$FinalPriceMin " /> </price>.

37. The method of claim 35, wherein Shipping cost see **FIG. 8** Line 28, 33 thru line 35 <freight type="Shipping is Selected mode in $persqft "> </freight>.
<xsl:value-of select="$FinalShipMin" /></freight>.

38. The method of claim 35, wherein Sales tax see **FIG. 8** Line 36 thru 38 <xsl:value-of select="($SalesTax_price div 100) * $FinalPriceMin * $productqty" /> </tax>.

39. The method of claim 35, wherein Other fields **FIG. 8** Line 22 <xsl:variable name="productDiscount" select=" $basePrice - ($basePrice * $productqtyDiscount) " />.

40. **File upload**: The input allows for the location of a local computer file to be uploaded to the server to a designated location. .

41. The method of claim 39, wherein **Upload URL**: Location on the web server directory or ftp or to your printer if your printer supports ftp, example c:/webhost4life aspnet/sriflcom/www_devstore/InputProductImages/ or ftp ftp://PSCFTPData:LogOnID@ftp.srifl.com/DataPath/ .

42. The method of claim 39, wherein **Upload temp URL**: The temp directory is specified as to where the file is placed until final check then it is put in the next location, example c:/webhost4life aspnet/sriflcom/www_devstore/InputProductImages/temp.

43. Indication of an element being a SVG type see **FIG. 3** Line 28 <isSVGFile>true</isSVGFile> . In **FIG. 7** is an example of what can be achieved. The novel aspect in this invention is the integrated retrieval of a

master SVG file then saving it with the new option variables and the pricing ability, real time design ability, and saving the results in a single database table.

44. Order final checkout will NOT occur until update cart has been successfully completed without any errors for processing integrity.

45. Each option is contained in an element definitions for standalone and for the Portal access additional subset <xs:annotation> within each element of the e-commerce system. The Portal will allow prospects to select by the subset options without forcing in house namespace labels as shown in **FIG. 11 field 1103**.

46. A system defined allowed number of suppliers can be selected to provide quotes via Web Services.

47. The returned quoted price is displayed to the prospect with the ability to select the supplier and award the contract.

48. The prospect continues on to the awarded supplier's e-commerce site for completion of credit processing information to consummate the commerce transaction.

49. The portal will retain XML information of the transactions in the database in a single database record in XML format.